UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/600,083	06/20/2003	Surajit Chaudhuri	301555.01	9011

22971 7590 05/04/2007

MICROSOFT CORPORATION
ONE MICROSOFT WAY
REDMOND, WA 98052-6399

EXAMINER	
HICKS, MICHAEL J	
ART UNIT	**PAPER NUMBER**
2165	

NOTIFICATION DATE	DELIVERY MODE
05/04/2007	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

roks@microsoft.com
ntovar@microsoft.com
a-rydore@microsoft.com

Office Action Summary	Application No.	Applicant(s)	
	10/600,083	CHAUDHURI ET AL.	
	Examiner	Art Unit	
	Michael J. Hicks	2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *27 October 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-19,21-37 and 39-46* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☒ Claim(s) *8, 12, 27, 30, 41, and 44* is/are allowed.

6)☒ Claim(s) *1-7, 9-11, 13-19, 21-26, 28-29, 31-37, 39-40, 42-43, and 45-46* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *20 June 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-19, 21-37, and 39-46 Pending.

Claims 20 and 38 Canceled.

Response to Arguments

2. Applicant's arguments, see response, filed 10/27/2006, with respect to the rejection(s) of claim(s) 1-46 under USC 102 and USC 103 have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of the reference of Kephart et al (U.S. Patent Number 7,051,277 and referred to hereinafter as Kephart) which cures the deficiencies of the previously existing art.

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-2, 10, and 13-16 rejected under 35 U.S.C. 103(a) as being unpatentable over Lepien (U.S. Patent Number 6,636,850) in view of Kephart.

As per Claims 1 and 16, Lepien discloses a process and system for testing an evaluation data record having attribute fields containing data (i.e. *"The present invention*

comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...Each of the comparisons is conducted on each record by examining each field n the transaction record and comparing the contents of the field to the contents of a corresponding record in one of a plurality of legacy records." The preceding text excerpt clearly indicates that a transaction record/evaluation data record having fields/attribute fields which contain contents/data is compared tested against one or more legacy records.) (Column 1, Lines 49-62) comprising: providing a reference table having a number of reference records against which a evaluation data record is tested (i.e. *" The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...These legacy records can exist, in for instance, a customer database."* The preceding text excerpt clearly indicates that the legacy records/reference records a re stored in a database, which further indicates that they are stored in a table/reference table in the database.) (Column 1, Lines 49-55); identifying reference table tokens contained within the reference records of the reference table (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match..."* The preceding text excerpt clearly indicates that the data in each field in the reference records is considered to be a token.) (Column 9, Lines 3-4) and determining a count of tokens in the reference table classified according to attribute field (i.e. *"The equal-7 comparison mechanism compares two fields after having converted both fields to their numeric equivalents."* The preceding text excerpt clearly indicates that each field/attribute field in both the legacy/reference records and the transaction/evaluation records has all of its data/tokens counted using their numeric value and recorded for comparison.) (Column 11, Lines 63-65); and assigning a similarity score to said evaluation data record in relation to a reference record within the reference table (i.e. *"The present invention is a merge or purge system that uses score-based matching condition between records."* The preceding text excerpt clearly indicates that comparisons (e.g. an evaluation data record in relation to a reference record) are assigned scores/similarity scores.) (Column 7, Lines 23-24) based on a

combination of: the number of common tokens of an evaluation field of the evaluation data record and a corresponding field within a reference record from the reference table (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match...The AL comparison mechanism compares two fields for a close alpha match...The letters E and O are treated as identical. The AL compare allows for one transposition of characters."* The preceding text excerpt clearly indicates that each datum contained in a field/token in the legacy/reference records is compared with each datum contained in a field/token in the transaction/evaluation record to find the number of common tokens (e.g. a common token indicates equality). The excerpt also demonstrates that the number of common tokens is determined and utilized in a number of different ways.) (Column 9, Lines 3-4; Column 13, Lines 11-17); and the similarity of the tokens that are not the same in the evaluation field of the evaluation data record and the corresponding field of the reference record from the reference table (i.e. *"The AL comparison mechanism compares two fields for a close alpha match...The letters E and O are treated as identical. The AL compare allows for one transposition of characters."* The preceding text excerpt clearly indicates that non-identical datum contained in fields/tokens is considered for their similarity (e.g. E and O are considered equivalent).) (Column 13, Lines 11-17).

Lepien fails to disclose a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table.

Kephart discloses a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table. (i.e. *"As discussed by Salton et al., direct comparison of the document's token frequencies with the token frequencies of each category can lead to highly inaccurate categorization because it tends to over-emphasize frequently occurring words such as "the" and "about." This problem is*

typically avoided by first converting the category token frequencies into category token weights that de-emphasize common words using the Term Frequency-Inverse Document Frequency (TF-IDF) principle. The TF-IDF weight for a token in a specific category increases with the frequency of that token among documents known to belong to the category and decreases with the frequency of that token within the entire collection of documents. There are many different TF-IDF weighting schemes. Salton et al. describe several weighting schemes and their implementations." The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field, which is based on the number of times the token appears in the category/reference data.) (Figure 10; Column 3, Lines 44-58).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Kephart to include a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table with the motivation of assisting a user with the task of categorizing a received electronic document into a collection (Kephart, Abstract).

As per Claim 2, Lepien discloses a look-up table based of contents of reference records in the reference table is prepared before evaluation of the evaluation data record (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name...During initialization, the initialization function requests a block of memory large enough to accommodate the match table...Each field name is used...to test a transaction record against one of a plurality of legacy records...Comparison of records is accomplished using the matching function. The matching function is called be the application program using three parameters being: reference to the parameter block; reference to the transaction record; and reference to the legacy records."* The preceding text excerpt clearly indicates that a matching table which is used by a matching function and contains reference to data in legacy records/look-up based table of contents of reference records is

prepared during initialization/before evaluation of records.) (Column 5, Lines 16-27; Column 7, Lines 13-18) and wherein the tokens of the evaluation data record are evaluated by comparing the contents of the look-up table with contents of the tokens of said evaluation data record to prepare a candidate set of reference records for which a similarity score is assigned (i.e. *"The matching function uses transaction reference parameter to retrieve the transaction record. Using a reference from the linked list of legacy records, the matching function then retrieves a legacy record...The matching function then performs a comparison of the first field specified in the matching table...When a field matches, the positive accumulator is incremented by the score value specified in the matching table for that field... Once all of the fields between two records have been compared, the matching function compares the value of the positive accumulator to a first positive threshold. If the positive accumulator exceeds the positive threshold a match is declared."* The preceding text excerpt clearly indicates that data from the matching table and the matching function/contents of the look-up table are used to compare datum in a field/tokens from the transaction/evaluation record to those of a legacy/reference record in order to assign a score for candidate records, a plurality of which may form a set.) (Column 8, Lines 9-36).

As per Claim 10, Lepien discloses reference records having a similarity score greater than a threshold are identified as candidate records (i.e. *"Once all of the fields between two records have been compared, the matching function compares the value of the positive accumulator to a first positive threshold. If the positive accumulator exceeds the positive threshold a match is declared. If a definitive match is not found, then the matching function compares the value stored in the negative accumulator to a second negative threshold. If the negative accumulator exceeds the negative threshold a mismatch is declared."* The preceding text excerpt clearly indicates that legacy/reference records, which have a match/similarity score which exceeds/is greater than a threshold, are identified as matches/candidates.) (Column 8, Lines 32-42).

As per Claim 13 Lepien discloses the tokens in different attribute fields are assigned different weights in determining said score (i.e. *"Because the matching system relies on scores, an implicit weighting of certain fields in the records can be helpful in confirming otherwise less than certain matches."* The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field (e.g. the tokens in the attribute field), and that the weights of different fields may vary.) (Column 7, Lines 29-31).

As per Claim 14, Lepien discloses assigning a score includes determining a cost in transposing the order of two tokens in determining a similarity between tokens of the input data record and records in the reference table (i.e. *"A very unique comparison mechanism that is tolerant of one character transposition...also comprises the present invention."* The preceding text excerpt clearly indicates that the cost of one transposition is acceptable, and the cost of two transpositions is unacceptable, therefore a cost of the transpositions is determined.) (Column 3, Lines 13-18).

As per Claims 15, Lepien discloses the determining of a cost in transposing tokens takes into account a weight of said tokens that are transposed (i.e. *"By assigning distinct scores for each score in a comparison, an implicit weighting of each field in any resultant aggregate score is achieved."* The preceding text excerpt clearly indicates that weight is taken into account, and is in fact built in, in all comparisons, which would include comparisons which determine the cost of a transposition.) (Column 5, Lines 38-41).

5. Claims 3, 9, and 17 rejected under 35 U.S.C. 103(a) as being unpatentable over Lepien in view Kephart and in further view of Califano (U.S. Patent Number 5,577,249).

As per Claims 3 and 17, Lepien and Kephart fail to disclose the preprocessor component evaluates tokens in the reference table by: breaking tokens in the reference table up into sets of substrings having a length q; applying a hash function to the set of substrings for a token to provide a vector representative of a token; and building a lookup table for substrings found within the tokens that make up the reference table.

Califano discloses the preprocessor component evaluates tokens in the reference table by: breaking tokens in the reference table up into sets of substrings having a length q (i.e. *"The method starts by selecting an original string from a database. The string is then partitioned into substrings of contiguous tokens...a number of original substrings of contiguous tokens are selected from an original token sequence in the database...The set members can all be a fixed number of tokens in length."* The preceding text excerpt clearly indicates that the original strings/tokens are broken up into substrings, which may all have length q (e.g. the same length).) (Column 5, Lines 32-35; Column 3, Lines 31-45); applying a hash function to the set of substrings for a token to provide a vector representative of a token (i.e. *"Using this set of original substring tokens, a set of tuples is formed...These tuples are called j-tuples where j is the number of original substrings which are used to form the tuple."* The preceding text excerpt clearly indicates that a hash function is used to form the set of substrings into a tuple which contains all of the substrings formed from the original string/a vector representation of the token. Note that while a hash function is not explicitly indicated, it would be natural to use a hash function to perform this operation.) (Column 3, Lines 37-46); and building a lookup table for substrings found within the tokens that make up the reference table (i.e. *"The original tuples are used to create original indexes which are then used to*

store information, associated with the index and the original string, in a cell in the look-up structure. This procedure is repeated for every original string of interest in the database...additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell." The preceding text excerpt clearly indicates that an index and look-up table are built for the tokens/vector representations which are constructed out of the substrings, and will contain location information for each individual substring.) (Column 5, Lines 36-40).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Califano to include the preprocessor component evaluates tokens in the reference table by: breaking tokens in the reference table up into sets of substrings having a length q; applying a hash function to the set of substrings for a token to provide a vector representative of a token; and building a lookup table for substrings found within the tokens that make up the reference table with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claim 9, Lepien and Kephart fail to disclose a closest K reference records from the reference table are identified as possible matches with the input record.

Califano discloses a closest K reference records from the reference table are identified as possible matches with the input record (i.e. *"The voting cells in the EIT which are accessed by the voting indexes are used to store 'votes' for an original string at a given match offset every time a corresponding match is registered by using the look-up structure and reference index as explained above. The value 'c' in each voting cell of the EIT is updates each time a voting index for that*

cell is generated. When a match occurs, i.e., a cell in the look-up structure has at least one information record, a voting index is generated, using the reference/pointer information record and the computer match offset." The preceding text excerpt clearly indicates that each time a match is indicated the voting cells are updated, which further indicates that each transaction/evaluation record may generate more than one match from the legacy/reference records.) (Column 12, Lines 62-67; Column 13, Lines 1-5).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Califano to include a closest K reference records from the reference table are identified as possible matches with the input record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

6. Claim 11 rejected under 35 U.S.C. 103(a) as being unpatentable over Lepien in view of Kephart and further in view of Califano (U.S. Patent Number 5,577,249) and Ananthakrishna et al. ("Eliminating Duplicates in Data Warehouses", Proceedings of the 28th International Conference on Very Large Databases (VLDB) 2002, Hong Kong and referred to hereinafter as Ananthakrishna).

As per Claim 11, Lepien discloses preparing the look-up table for tokens that make up the reference table by creating an entry in the look-up table for a token including an attribute field for the token or a substring (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text

excerpt clearly indicates that the matching table/lookup table includes a field name.) (Column 5, Lines 16-27).

Lepien and Kephart fail to disclose a step of evaluating tokens in the reference table by applying a function to the set of substrings for a token to provide a vector representative of a token, and the look-up table further comprising an attribute field for a co-ordinate within a vector for said token or substring, the look-up table further comprising an attribute field for a frequency of said token or substring, and a list of reference records where said token or said substring appears in the specified field and vector co-ordinate position

Califano discloses a step of evaluating tokens in the reference table by applying a function to the set of substrings for a token to provide a vector representative of a token (i.e. *"Using this set of original substring tokens, a set of tuples is formed...These tuples are called j-tuples where j is the number of original substrings which are used to form the tuple."* The preceding text excerpt clearly indicates that a hash function is used to form the set of substrings into a tuple which contains all of the substrings formed from the original string/a vector representation of the token. Note that while a hash function is not explicitly indicated, it would be natural to use a hash function to perform this operation.) (Column 3, Lines 37-46); and the look-up table further comprising an attribute field for a co-ordinate within a vector for said token or substring (i.e. *"...More preferably, the information record will also contain information about the location on the original string of the original tuple. Even more preferably, additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell."* The preceding text excerpt clearly indicates that location information about the tuples/vectors and location information/coordinates about the substrings used to create those tuples/vectors is included in the look-up table.) (Column 9, Lines 40-47).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Califano to include a step of evaluating tokens in the reference table by applying a function to the set of substrings for a token to provide a vector representative of a token and the look-up table further comprising an attribute field for a co-ordinate within a vector for said token or substring with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

Ananthakrishna discloses the look-up table further comprising an attribute field for a frequency of said token or substring (i.e. *"We build a token table of G containing the following information...the frequencies of said tokens."* The preceding text excerpt clearly indicates that the token/lookup table includes token frequencies.) (Page 6, Column 2, Paragraph 6), and a list of reference records where said token or said substring appears in the specified field and vector co-ordinate position (i.e. *"We build a token table of G containing the following information...the list of (pointers to) tuples in which such a token occurs...In figure 1, suppose we are processing the State relation grouped with the Country relation and that we detected the set...to be duplicates on the Country relation."* The preceding text excerpt clearly indicates that the token/lookup table includes a list of tuples/vectors in which the token/substring occurs. Note that the text expert in relation to Figure 1 further indicates that the token/substring must also appear in the specified attribute field, in this case 'Country', and have the specified vector coordinate position.) (Figure 1; Page 6, Column 2, Paragraph 6; Page 7, Column 1, Paragraph 2).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Ananthakrishna to include the entries further comprise: a frequency of said substring and a list of reference records where said substring appears in the specified attribute field and vector co-ordinate position with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

7. Claims 4-7, 18-26, 28-29, 21-40, 42-43, and 45-46 rejected under 35 U.S.C. 103(a) as being unpatentable over Lepien in view of Kephart further in view of Califano (U.S. Patent Number 5,577,249) and further in view of Ananthakrishna et al. ("Eliminating Duplicates in Data Warehouses", Proceedings of the 28th International Conference on Very Large Databases (VLDB) 2002, Hong Kong and referred to hereinafter as Ananthakrishna).

As per Claims 4 and 18, Lepien discloses the process of building the lookup table creates an entry comprising: an attribute field (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text excerpt clearly indicates that the matching table/lookup table includes a field name.) (Column 5, Lines 16-27).

Lepien and Kephart fail to disclose that the entries are for substrings and that the entries further comprise: a co-ordinate within a vector for said substring, a frequency of said substring, and a list of reference records where said substring appears in the specified attribute field and vector co-ordinate position.

Califano discloses the entries are for substrings (i.e. *"The original tuples are used to create original indexes which are then used to store information, associated with the index and the*

original string, in a cell in the look-up structure. This procedure is repeated for every original string of interest in the database...additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell." The preceding text excerpt clearly indicates that an index and look-up table are built for the tokens/vector representations which are constructed out of the substrings, and will contain location information for each individual substring.) (Column 5, Lines 36-40) and that the entries further comprise: a co-ordinate within a vector for said substring (i.e. *"...More preferably, the information record will also contain information about the location on the original string of the original tuple. Even more preferably, additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell."* The preceding text excerpt clearly indicates that location information about the tuples/vectors and location information/coordinates about the substrings used to create those tuples/vectors is included in the look-up table.) (Column 9, Lines 40-47).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Califano to include the entries are for substrings and that the entries further comprise: a co-ordinate within a vector for said substring with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

Ananthakrishna discloses that the entries further comprise: a frequency of said substring (i.e. *"We build a token table of G containing the following information...the frequencies of said tokens."* The preceding text excerpt clearly indicates that the token/lookup table includes token frequencies.) (Page 6, Column 2, Paragraph 6), and a list of reference records where said substring appears in the specified attribute field and vector co-ordinate position (i.e. *"We*

build a token table of G containing the following information...the list of (pointers to) tuples in which such a token occurs...In figure 1, suppose we are processing the State relation grouped with the Country relation and that we detected the set...to be duplicates on the Country relation." The preceding text excerpt clearly indicates that the token/lookup table includes a list of tuples/vectors in which the token/substring occurs. Note that the text expert in relation to Figure 1 further indicates that the token/substring must also appear in the specified attribute field, in this case 'Country', and have the specified vector coordinate position.) (Figure 1; Page 6, Column 2, Paragraph 6; Page 7, Column 1, Paragraph 2).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Ananthakrishna to include the entries further comprise: a frequency of said substring and a list of reference records where said substring appears in the specified attribute field and vector co-ordinate position with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

As per Claim 5, Lepien discloses the weights that are assigned to tokens of the evaluation record are distributed across candidate records from the reference table during a determination of a candidate set of records (i.e. *"By assigning distinct scores for each score in a comparison, an implicit weighting of each field in any resultant aggregate score is achieved."* The preceding text excerpt clearly indicates that because the weights are utilized in assigning the scores to candidate records, the weights must have been distributed across the candidate records from the reference table as they were determined.) (Column 5, Lines 38-41).

As per Claims 6 and 39, Lepien and Kephart fail to disclose a candidate record table is built and records listed in the lookup table are added to the candidate record table based on vector representations of the tokens of the input record.

Califano discloses a candidate record table is built (i.e. *"In summary, after the exactly and similarly matching original strings have been determined, they are located in the database."* The preceding text excerpt clearly indicates the matching candidate strings/candidate records are located in the database. Note that once the original string is located, its associated record is also located. Also note that in order to track these matching strings for further processing, a table of the candidate records would have to be built to store them.) (Column 14, Lines 9-20) and records listed in the lookup table are added to the candidate record table based on vector representations of the tokens of the input record (i.e. *"It is also possible to locate the matching string and sequence(s) using the information record in the look-up table in the cell which caused the vote."* The preceding text excerpt clearly indicates that matching records in the look-up table may be located/added to the candidate record table using/based on the information record in the look-up table which includes the sequences of tokens/vector representation of the tokens of the input record.) (Column 10, Lines 49-55; Column 14, Lines 18-20).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Califano to include a candidate record table is built and records listed in the lookup table are added to the candidate record table based on vector representations of the tokens of the input record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claims 7, 26 and 40, Lepien and Kephart fail to disclose a candidate record is added to the candidate record table only if a score assigned to the reference record can exceed a threshold based on an already evaluated substring.

Califano discloses a candidate record is added to the candidate record table only if a score assigned to the reference record can exceed a threshold based on an already evaluated substring (i.e. *"...all of the cells in the EIT having a value of c above a given threshold are selected as indications of matching original strings. Then a rating of similarity, which is directly proportional to the value c, can be computed....Therefore, by comparing the number of cotes in the EIT that different original strings receive when compared to a given reference string, a degree of similarity between each original string and the reference record can be established. The original strings receiving a higher number of votes in the EIT (after all reference tuples are compared) are more similar to the reference string that original strings receiving a lower number of votes."* The preceding text excerpt clearly indicates that the candidate record/reference record is only designated as a match (and therefore added to the candidate record table) only if a similarity score which is assigned to it exceeds a threshold based on at least one already evaluated substring.) (Column 13, Lines 37-41; Column 14, Lines 1-8).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien and Kephart with the teachings of Califano to include a candidate record is added to the candidate record table only if a score assigned to the reference record can exceed a threshold based on an already evaluated substring with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claim 19, Lepien discloses a process for evaluating an input data record having attribute fields containing data (i.e. *"The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...Each of the comparisons is conducted on each record by examining each field n the transaction record and comparing the contents of the field to the contents of a corresponding record in one of a plurality of legacy records."* The preceding text excerpt clearly indicates that a transaction record/input data record having fields/attribute fields which contain contents/data is compared tested against one or more legacy records.) (Column 1, Lines 49-62) comprising: providing a number of reference records organized into attribute fields against which an input data record is evaluated (i.e. *" The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...These legacy records can exist, in for instance, a customer database."* The preceding text excerpt clearly indicates that the legacy records/reference records a re stored in a database, which further indicates that they are stored in a table/reference table in the database.) (Column 1, Lines 49-55), evaluating reference records to identify tokens from said attribute fields (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match..."* The preceding text excerpt clearly indicates that the data in each field in the reference records is considered to be a token.) (Column 9, Lines 3-4), looking up reference records in the index table based on the contents of the input record and selecting a number of candidate records from the reference records in the index table for comparing to said input data record. (i.e. *"The matching function uses transaction reference parameter to retrieve the transaction record. Using a reference from the linked list of legacy records, the matching function then retrieves a legacy record...The matching function then performs a comparison of the first field specified in the matching table...When a field matches, the positive accumulator is incremented by the score value specified in the matching table for that field... Once all of the fields*

between two records have been compared, the matching function compares the value of the positive accumulator to a first positive threshold. If the positive accumulator exceeds the positive threshold a match is declared." The preceding text excerpt clearly indicates that data from the matching table and the matching function/contents of the reference table are used to compare datum in a field/tokens from the transaction/evaluation record to those of a legacy/reference record in order to select a number of candidate records, a plurality of which may form a set.) (Column 8, Lines 9-36), and assigning a similarity score to said evaluation data record in relation to a candidate set of reference records (i.e. *"The present invention is a merge or purge system that uses score-based matching condition between records."* The preceding text excerpt clearly indicates that comparisons (e.g. an evaluation data record in relation to a reference record) are assigned scores/similarity scores.) (Column 7, Lines 23-24) based on a combination of: the number of common tokens of an evaluation field of the input data record and a corresponding field within a reference record (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match...The AL comparison mechanism compares two fields for a close alpha match...The letters E and O are treated as identical. The AL compare allows for one transposition of characters."* The preceding text excerpt clearly indicates that each character/token in the legacy/reference records is compared with each character/token in the transaction/evaluation record to find the number of common tokens. The excerpt also demonstrates that the number of common token is taken into account in a number of different ways.) (Column 9, Lines 3-4; Column 13, Lines 11-17); the similarity of the tokens that are not the same in the evaluation field of the input data record and the corresponding field of the reference record (i.e. *"The AL comparison mechanism compares two fields for a close alpha match...The letters E and O are treated as identical. The AL compare allows for one transposition of characters."* The preceding text excerpt clearly indicates that non-identical characters/tokens are considered for their similarity (e.g. E and O are considered equivalent).) (Column 13, Lines 11-17).

Lepien fails to disclose evaluating each token to build a vector of token substrings that represent the token; building an index table wherein entries of the index table contains a token substring, a list of reference records that contain a token that maps to the token substring, and a weight of the tokens in the evaluation field of the input data record based on a count of the tokens from the corresponding field contained within the reference records.

Califano discloses evaluating each token to build a vector of token substrings that represent the token (i.e. *"The method starts by selecting an original string from a database. The string is then partitioned into substrings of contiguous tokens...a number of original substrings of contiguous tokens are selected from an original token sequence in the database...The set members can all be a fixed number of tokens in length...Using this set of original substring tokens, a set of tuples is formed..."* The preceding text excerpt clearly indicates that the original strings/ reference tokens are broken up into substrings which is then represented by a tuple/vector.) (Column 5, Lines 32-35; Column 3, Lines 31-46).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include evaluating each token to build a vector of token substrings that represent the token with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

Ananthakrishna discloses building an index table wherein entries of the index table contains a token substring and a list of reference records that contain a token that maps to the token substring (i.e. *"We build a token table of G containing the following*

information...the set of tokens whose frequency...is greater than one...the list of (pointers to) tuples in which such a token occurs..." The preceding text excerpt clearly indicates that the token/lookup table includes a list of tuples/vectors in which the token/substring occurs (e.g. which maps to the token/substring).) (Page 6, Column 2, Paragraph 6).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Ananthakrishna to include building an index table wherein entries of the index table contains a token substring and a list of reference records that contain a token that maps to the token substring with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

Kephart discloses a weight of the tokens in the evaluation field of the input data record based on a count of the tokens from the corresponding field contained within the reference records. (i.e. *"As discussed by Salton et al., direct comparison of the document's token frequencies with the token frequencies of each category can lead to highly inaccurate categorization because it tends to over-emphasize frequently occurring words such as "the" and "about." This problem is typically avoided by first converting the category token frequencies into category token weights that de-emphasize common words using the Term Frequency-Inverse Document Frequency (TF-IDF) principle. The TF-IDF weight for a token in a specific category increases with the frequency of that token among documents known to belong to the category and decreases with the frequency of that token within the entire collection of documents. There are many different TF-IDF weighting schemes. Salton et al. describe several weighting schemes and their implementations."* The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field, which is based on the number of times the token appears in the category/reference data.) (Figure 10; Column 3, Lines 44-58).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Kephart to include a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table with the motivation of assisting a user with the task of categorizing a received electronic document into a collection (Kephart, Abstract).

As per Claim 21, Lepien fails to disclose a candidate record table is built and candidate records from the index table are added to a candidate record table based on an H dimensional vector of token substrings determined from tokens contained in the input record.

Califano discloses a candidate record table is built (i.e. *"In summary, after the exactly and similarly matching original strings have been determined, they are located in the database."* The preceding text excerpt clearly indicates the matching candidate strings/candidate records are located in the database. Note that once the original string is located, its associated record is also located. Also note that in order to track these matching strings for further processing, a table of the candidate records would have to be built to store them.) (Column 14, Lines 9-20) and candidate records from the index table are added to a candidate record table based on an H dimensional vector of token substrings determined from tokens contained in the input record (i.e. *"It is also possible to locate the matching string and sequence(s) using the information record in the look-up table in the cell which caused the vote."* The preceding text excerpt clearly indicates that matching records in the look-up table/index may be located/added to the candidate record table using/based on the information record in the look-up table which includes the sequences of tokens/an H dimensional vector of token

substrings determined from tokens of the input record. Note that an H dimensional vector could be a one dimensional vector.) (Column 10, Lines 49-55; Column 14, Lines 18-20).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include a candidate record table is built and candidate records from the index table are added to a candidate record table based on an H dimensional vector of token substrings determined from tokens contained in the input record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claim 22, Lepien fails to disclose that tokens are parsed from the input data record and tokens contained in said input data record are assigned token weights based on occurrences of the tokens in the reference table and further wherein records added to the candidate record table are factored by an amount corresponding to the weights of tokens extracted from the input data record.

Califano discloses that tokens are parsed from the input data record and tokens contained in said input data record are assigned token weights based on occurrences of the tokens in the reference table (i.e. *"The voting cells in the EIT, which are accesses by the voting indexes are used to store 'votes' for an original string at a given match offset every time a corresponding match is registered by the look-up structure and reference index as explained above. The value 's' in each voting cell of the ET is updated each time a voting index for that cell is generated."* The preceding text excerpt clearly indicates that input data records and their corresponding tokens are assigned votes/a

weight which corresponds to the number of matches the records or token has in the reference index/table.) (Column 12, Lines 62-67; Column 13, Line 1) and further wherein records added to the candidate record table are factored by an amount corresponding to the weights of tokens extracted from the input data record (i.e. *"...all of the cells in the EIT having a value of c above a given threshold are selected as indications of matching original strings. Then a rating of similarity, which is directly proportional to the value c, can be computed."* The preceding text excerpt clearly indicates that the candidate records (e.g. those having a value of c above the threshold) are factored by an amount directly proportional/corresponding to their number of votes/weight.) (Column 13, Lines 37-41).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include that tokens are parsed from the input data record and tokens contained in said input data record are assigned token weights based on occurrences of the tokens in the reference table and further wherein records added to the candidate record table are factored by an amount corresponding to the weights of tokens extracted from the input data record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claim 23, Lepien discloses weights are assigned to tokens based on the attribute field in which the tokens are contained in the reference table (i.e. *"Because the matching system relies on scores, an implicit weighting of certain fields in the records can be helpful in confirming otherwise less than certain matches."* The preceding text excerpt clearly indicates that a

weight is assigned to the comparison data for a particular field (e.g. the tokens in the attribute field) of the reference table.) (Column 7, Lines 29-31).

As per Claim 24, Lepien discloses a step of assigning a similarity score to said input data record in relation to a candidate set of reference records based on: a cost in converting tokens in the input data record to tokens in a corresponding field of a reference record (i.e. *"A very unique comparison mechanism that is tolerant of one character transposition...also comprises the present invention."* The preceding text excerpt clearly indicates that a cost is considered for converting the input data tokens to tokens in a corresponding field of a reference record (e.g. the cost of one character transposition is acceptable, but the cost of more than one character transposition is not).) (Column 3, Lines 13-18) wherein the cost is based on a weight of the tokens in the corresponding field of said reference record corresponding to a count of the tokens from the corresponding field contained within the reference records (i.e. *" By assigning distinct scores for each score in a comparison, an implicit weighting of each field in any resultant aggregate score is achieved... Because the matching system relies on scores, an implicit weighting of certain fields in the records can be helpful in confirming otherwise less than certain matches."* The preceding text excerpt clearly indicates that weight is taken into account, and is in fact built in, in all comparisons, which would include comparisons which determine the cost of a transposition. Note that the comparison data for a particular field on which the weight is based may include a count of the tokens as disclosed above.) (Column 5, Lines 38-41; Column 7, Lines 29-31).

As per Claim 25, Lepien discloses the reference records are stored in a reference table (i.e. *" The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...These legacy records*

can exist, in for instance, a customer database." The preceding text excerpt clearly indicates that the legacy records/reference records a re stored in a database, which further indicates that they are stored in a table/reference table in the database.) (Column 1, Lines 49-55).

Lepien fails to disclose a candidate record table is built and candidate records from the index table are added to a candidate record table based on token substrings contained in the input record and wherein tokens contained in said input data record are assigned token weights based on occurrences of the tokens in the reference table and further wherein records added to the candidate record table are factored by an amount corresponding to the weights of tokens contained in the input data record.

Califano discloses a candidate record table is built (i.e. *"In summary, after the exactly and similarly matching original strings have been determined, they are located in the database."* The preceding text excerpt clearly indicates the matching candidate strings/candidate records are located in the database. Note that once the original string is located, its associated record is also located. Also note that in order to track these matching strings for further processing, a table of the candidate records would have to be built to store them.) (Column 14, Lines 9-20) and candidate records from the index table are added to a candidate record table based on token substrings contained in the input record (i.e. *"It is also possible to locate the matching string and sequence(s) using the information record in the look-up table in the cell which caused the vote."* The preceding text excerpt clearly indicates that matching records in the look-up table may be located/added to the candidate record table using/based on the information record in the look-up table which includes the token substrings contained in the input record.) (Column 10, Lines 49-55; Column 14, Lines 18-20) and wherein tokens contained in said input data record are assigned token weights based on occurrences of the tokens in the reference table (i.e. *"The voting cells in the EIT which are accessed by the voting indexes are used to store 'votes' for an original string at a given match offset every time a corresponding*

match is registered by using the look-up structure and reference index as explained above. The value 'c' in each voting cell of the EIT is updates each time a voting index for that cell is generated." The preceding text excerpt clearly indicates that each time a match a token of the input record and a token in the reference table is indicated the tokens are assigned another vote, which acts as a weights (e.g. the more votes a token receives, the higher it's weight).) (Column 12, Lines 62-67) and further wherein records added to the candidate record table are factored by an amount corresponding to the weights of tokens contained in the input data record (i.e. *"...all of the cells in the EIT having a value of c above a given threshold are selected as indications of matching original strings. Then a rating of similarity, which is directly proportional to the value c, can be computed."* The preceding text excerpt clearly indicates that the candidate records (e.g. the records whose value of c meets the threshold criteria) are then assigned a similarity score/factored by an amount which is directly proportional/corresponding to the number of votes/weight.) (Column 13, Lines 37-41).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include a candidate record table is built and candidate records from the index table are added to a candidate record table based on token substrings contained in the input record and wherein tokens contained in said input data record are assigned token weights based on occurrences of the tokens in the reference table and further wherein records added to the candidate record table are factored by an amount corresponding to the weights of tokens contained in the input data record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claims 28 and 42, Lepien fails to disclose a closest K reference records from the reference table are identified as possible matches with the input record.

Califano discloses a closest K reference records from the reference table are identified as possible matches with the input record (i.e. *"The voting cells in the EIT which are accessed by the voting indexes are used to store 'votes' for an original string at a given match offset every time a corresponding match is registered by using the look-up structure and reference index as explained above. The value 'c' in each voting cell of the EIT is updates each time a voting index for that cell is generated. When a match occurs, i.e., a cell in the look-up structure has at least one information record, a voting index is generated, using the reference/pointer information record and the computer match offset."* The preceding text excerpt clearly indicates that each time a match is indicated the voting cells are updated, which further indicates that each transaction/evaluation record may generate more than one match from the legacy/reference records.) (Column 12, Lines 62-67; Column 13, Lines 1-5).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include a closest K reference records from the reference table are identified as possible matches with the input record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

As per Claim 29, Lepien discloses reference records having a similarity score greater than a threshold are identified as candidate records (i.e. *"Once all of the fields between two records have been compared, the matching function compares the value of the positive accumulator to a first positive threshold. If the positive accumulator exceeds the positive threshold a*

match is declared. If a definitive match is not found, then the matching function compares the value stored in the negative accumulator to a second negative threshold. If the negative accumulator exceeds the negative threshold a mismatch is declared." The preceding text excerpt clearly indicates that legacy/reference records, which have a match/similarity score which exceeds/is greater than a threshold, are identified as matches/candidates.) (Column 8, Lines 32-42).

As per Claim 31, Lepien discloses the tokens in different attribute fields are assigned different weights in determining said score (i.e. *"Because the matching system relies on scores, an implicit weighting of certain fields in the records can be helpful in confirming otherwise less than certain matches."* The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field (e.g. the tokens in the attribute field), and that the weights of different fields may vary.) (Column 7, Lines 29-31).

As per Claims 32, 36, and 46 Lepien discloses each entry of the index table additionally comprises an attribute field for the token from which a substring is derived (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text excerpt clearly indicates that the matching table/lookup table includes a field name.) (Column 5, Lines 16-27).

As per Claim 33, Lepien discloses the index table entries also contain an attribute field (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text excerpt clearly indicates that the matching table/index table includes is built and includes a field/attribute name.) (Column 5, Lines 16-27).

Lepien fails to disclose the vector is an H dimensional vector of token substrings, and the index table entries also contain a position within the H dimensional vector, and a frequency of reference records that map to the token substring contained in an index table entry.

Califano discloses the vector is an H dimensional vector of token substrings (i.e. *"Using this set of original substring tokens, a set of tuples is formed...These tuples are called j-tuples where j is the number of original substrings which are used to form the tuple."* The preceding text excerpt clearly indicates that a function is used to form the set of substrings into a tuple which contains all of the substrings formed from the original string/an H dimensional vector of token substrings. Note that an H dimensional vector could be a one dimensional vector.) (Column 3, Lines 37-46) and the index table entries also contains a position within the H dimensional vector (i.e. *"...More preferably, the information record will also contain information about the location on the original string of the original tuple. Even more preferably, additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell."* The preceding text excerpt clearly indicates that location information about the tuples/vectors and location information of/the position within an H dimensional vector based on said token is included in the index.) (Column 9, Lines 40-47).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include the vector is an H dimensional vector of token substrings and the index table entries also contains a position within the H dimensional vector with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

Ananthakrishna discloses the index table entries also contains a frequency of reference records that map to the token substring contained in an index table entry (i.e. *"We build a token table of G containing the following information...the frequencies of said tokens."* The preceding text excerpt clearly indicates that the token/lookup table includes token frequencies of records which contain/map to the token substring in the index entry.) (Page 6, Column 2, Paragraph 6).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Ananthakrishna to include the index table entries also contains a frequency of reference records that map to the token substring contained in an index table entry with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

As per Claim 34, Lepien discloses a system for evaluating an input data record having fields containing data (i.e. *"The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...Each of the comparisons is conducted on each record by examining each field n the transaction record and comparing the contents of the field to the contents of a corresponding record in one of a plurality of legacy records."* The preceding text excerpt clearly indicates that a transaction record/input data record having fields/attribute fields which contain contents/data is compared tested against one or more legacy records.) (Column 1, Lines 49-62) comprising: a database for storing a reference table having a number of reference records against which an input data record is evaluated (i.e. *" The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...These legacy records can exist, in for instance, a customer database."* The preceding text excerpt clearly indicates that the legacy records/reference

records a re stored in a database, which further indicates that they are stored in a table/reference table in the database and may be used to evaluate transaction/input data records..) (Column 1, Lines 49-55); a preprocessor component for evaluating reference records in the reference table to identify tokens (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match..."* The preceding text excerpt clearly indicates that the data in each field in the reference records is considered to be a token.) (Column 9, Lines 3-4) and determining a count of tokens in the reference table classified according to record field (i.e. *"The equal-7 comparison mechanism compares two fields after having converted both fields to their numeric equivalents."* The preceding text excerpt clearly indicates that each field/attribute field in both the legacy/reference records and the transaction/input data records has all of its data/tokens counted using their numeric value and recorded for comparison.) (Column 11, Lines 63-65); said preprocessor evaluating reference records to identify tokens from said attribute fields (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match..."* The preceding text excerpt clearly indicates that the data in each attribute field in the reference records is considered to be a token.) (Column 9, Lines 3-4), entries of the index table contain an attribute field (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text excerpt clearly indicates that the matching table/index table includes is built and includes a field/attribute name.) (Column 5, Lines 16-27) and assigning a score to said candidate records (i.e. *"Because the matching system relies on scores, an implicit weighting of certain fields in the records can be helpful in confirming otherwise less than certain matches."* The preceding text excerpt clearly indicates that a score, which is affected by a weight.) (Column 7, Lines 29-31).

Lepien fails to disclose evaluating each token to build a H dimensional vector of token substrings that represent the token; and building an index table wherein entries of

the index table contains a token substring, a position within the H dimensional vector, and a matching component for assigning a score to an input data record in relation to a reference record within the reference table by building a candidate record table of candidate records from the index table based on an H dimensional vector of token substrings determined from tokens contained in the input record, entries of the index table contain a list of reference records, and the assigned score is based on a weight of the tokens of the input data record that is based on a count of the tokens from the corresponding field contained within the reference table.

Califano discloses evaluating each token to build a H dimensional vector of token substrings that represent the token (i.e. *"Using this set of original substring tokens, a set of tuples is formed...These tuples are called j-tuples where j is the number of original substrings which are used to form the tuple."* The preceding text excerpt clearly indicates that a function is used to form the set of/evaluate each substrings/token into a tuple which contains all of the substrings formed from the original string/an H dimensional vector of token substrings. Note that an H dimensional vector could be a one dimensional vector.) (Column 3, Lines 37-46); and building an index table wherein entries of the index table contains a token substring (i.e. *"The original tuples are used to create original indexes which are then used to store information, associated with the index and the original string, in a cell in the look-up structure. This procedure is repeated for every original string of interest in the database...additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell."* The preceding text excerpt clearly indicates that an index and look-up table are built for the tokens/vector representations which are constructed out of the substrings (and therefore include substrings), and entries will contain location information for each individual substring as well as the substrings themselves.) (Column 5, Lines 36-40), and a position within the H dimensional vector (i.e. *"...More preferably, the information record will also contain information about*

the location on the original string of the original tuple. Even more preferably, additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell." The preceding text excerpt clearly indicates that location information about the tuples/vectors and location information of/the position within an H dimensional vector based on said token is included in the index.) (Column 9, Lines 40-47), and a matching component for assigning a score to an input data record in relation to a reference record within the reference table (i.e. *"...all of the cells in the EIT having a value of c above a given threshold are selected as indications of matching original strings. Then a rating of similarity, which is directly proportional to the value c, can be computed."* The preceding text excerpt clearly indicates that the original/input data records are assigned a similarity score/score in relation to reference records.) (Column 13, Lines 37-41) by building a candidate record table of candidate records from the index table based on an H dimensional vector of token substrings determined from tokens contained in the input record (i.e. *"In summary, after the exactly and similarly matching original strings have been determined, they are located in the database...It is also possible to locate the matching original string and sequence(s using the information record in the look-up table in the cell which caused the vote.)"* The preceding text excerpt clearly indicates the matching candidate strings/candidate records are located in the database based on information in the look-up table, which includes the H-dimensional vector of token substrings determined from tokens in the input data record. Note that once the original string is located, its associated record is also located. Also note that in order to track these matching strings for further processing, a table of the candidate records would have to be built to store them.) (Column 14, Lines 9-20).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include evaluating each token to build a H dimensional vector of token substrings that represent the token; and building an index table wherein entries of the index table contains a token substring, a position within the H dimensional vector, and a matching component

for assigning a score to an input data record in relation to a reference record within the reference table by building a candidate record table of candidate records from the index table based on an H dimensional vector of token substrings determined from tokens contained in the input record with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49)

Ananthakrishna discloses entries of the index table contain a list of reference records (i.e. *"We build a token table of G containing the following information...the list of (pointers to) tuples in which such a token occurs... "* The preceding text excerpt clearly indicates that the token/index table includes a list of tuples/reference records in which the token/substring occurs.) (Page 6, Column 2, Paragraph 6).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Ananthakrishna to include entries of the index table contain a list of reference records with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

Kephart discloses the assigned score is based on a weight of the tokens of the input data record that is based on a count of the tokens from the corresponding field contained within the reference table (i.e. *"As discussed by Salton et al., direct comparison of the document's token frequencies with the token frequencies of each category can lead to highly inaccurate categorization because it tends to over-emphasize frequently occurring words such as "the" and "about." This problem is typically avoided by first converting the category token frequencies into category token*

weights that de-emphasize common words using the Term Frequency-Inverse Document Frequency (TF-IDF) principle. The TF-IDF weight for a token in a specific category increases with the frequency of that token among documents known to belong to the category and decreases with the frequency of that token within the entire collection of documents. There are many different TF-IDF weighting schemes. Salton et al. describe several weighting schemes and their implementations." The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field, which is based on the number of times the token appears in the category/reference data.) (Figure 10; Column 3, Lines 44-58).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Kephart to include the assigned score is based on a weight of the tokens of the input data record that is based on a count of the tokens from the corresponding field contained within the reference table with the motivation of assisting a user with the task of categorizing a received electronic document into a collection (Kephart, Abstract).

As per Claim 35, Lepien discloses a data structure encoded on a computer readable medium for use in evaluating an input data record having fields containing data comprising: a reference table organized in attribute columns having a number of records against which an input data record is evaluated (i.e. *" The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...These legacy records can exist, in for instance, a customer database."* The preceding text excerpt clearly indicates that the legacy records/reference records which are organized into attribute fields are stored in a database, which further indicates that they are stored in a table/reference table in the database. The excerpt also indicates that a transaction/input record is evaluated against these records.) (Column 1, Lines 49-55); each entry in the index table includes

a column of the reference table having said token from which the token is derived (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text excerpt clearly indicates that the matching table/index table includes a field name/column of the reference table having said token from which the token is derived.) (Column 5, Lines 16-27) and assigning a similarity score to said evaluation data record in relation to a reference record within the reference table (i.e. *"The present invention is a merge or purge system that uses score-based matching condition between records."* The preceding text excerpt clearly indicates that comparisons (e.g. an evaluation data record in relation to a reference record) are assigned scores/similarity scores.) (Column 7, Lines 23-24).

Lepien fails to disclose an index table wherein each entry of the index table contains a token substring from a token in the reference table, each entry in the index table includes a position within a H dimensional vector based on said token and the score is based on a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table.

Califano discloses an index table wherein each entry of the index table contains a token substring from a token in the reference table (i.e. *"The index generating algorithm is used to create a unique original index for each tuple in the set of original tuples...formed from the original substrings..."* The preceding text excerpt clearly indicates that each entry of the index contains a tuple, which further has a token substring.) (Column 9, Lines 8-10), and each entry in the index table includes a position within a H dimensional vector based on said token (i.e. *"...More preferably, the information record will also contain information about the location on the original string of the original tuple. Even more preferably, additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell."* The preceding text excerpt clearly

indicates that location information about the tuples/vectors and location information of/the position within an H dimensional vector based on said token is included in the index.) (Column 9, Lines 40-47).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include an index table wherein each entry of the index table contains a token substring from a token in the reference table and each entry in the index table includes a position within a H dimensional vector based on said token the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

Ananthakrishna discloses an index table wherein each entry of the index table contains a token substring from a token in the reference table and a list of records contained within the reference table (i.e. *"We build a token table of G containing the following information...the set of tokens whose frequency...is greater than one...the list of (pointers to) tuples in which such a token occurs..."* The preceding text excerpt clearly indicates that the token/index table includes a list of tuples/vectors in which the token/substring occurs (e.g. which maps to the token/substring), as well as the token substring from the reference table.) (Page 6, Column 2, Paragraph 6).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Ananthakrishna to include an index table wherein each entry of the index table contains a token substring from a token in the reference table and a list of records contained within the reference

table with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

Kephart discloses the score is based on a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table. (i.e. *"As discussed by Salton et al., direct comparison of the document's token frequencies with the token frequencies of each category can lead to highly inaccurate categorization because it tends to over-emphasize frequently occurring words such as "the" and "about." This problem is typically avoided by first converting the category token frequencies into category token weights that de-emphasize common words using the Term Frequency-Inverse Document Frequency (TF-IDF) principle. The TF-IDF weight for a token in a specific category increases with the frequency of that token among documents known to belong to the category and decreases with the frequency of that token within the entire collection of documents. There are many different TF-IDF weighting schemes. Salton et al. describe several weighting schemes and their implementations."* The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field, which is based on the number of times the token appears in the category/reference data.) (Figure 10; Column 3, Lines 44-58).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Kephart to include the score is based on a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table with the motivation of assisting a user with the task of categorizing a received electronic document into a collection (Kephart, Abstract).

As per Claim 37, Lepien discloses a machine readable medium including instructions for evaluating an input data record having attribute fields (i.e. *"The present*

invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...Each of the comparisons is conducted on each record by examining each field n the transaction record and comparing the contents of the field to the contents of a corresponding record in one of a plurality of legacy records." The preceding text excerpt clearly indicates that a transaction record/input data record having fields/attribute fields which contain contents/data is compared tested against one or more legacy records.) (Column 1, Lines 49-62) containing by steps of: accessing a reference table having a number of records organized into attribute fields against which an input data record is evaluated (i.e. *" The present invention comprises a method and apparatus that allows for a flexible comparison of a transaction record to a plurality of known data records...These legacy records can exist, in for instance, a customer database."* The preceding text excerpt clearly indicates that the legacy records/reference records which are organized into attribute fields are stored in a database, which further indicates that they are stored in a table/reference table in the database. The excerpt also indicates that a transaction/input record is evaluated against these records.) (Column 1, Lines 49-55); evaluating records in the reference table to identify tokens from said attribute fields (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match..."* The preceding text excerpt clearly indicates that the data in each field in the reference records is considered to be a token.) (Column 9, Lines 3-4); building an index table wherein each entry of the index table contains a column of the reference table (i.e. *"The matching table comprises a plurality of records each of which further comprises the following: field name..."* The preceding text excerpt clearly indicates that the matching table/index table includes is built and includes a field/column of the reference table name.) (Column 5, Lines 16-27); looking up records in the index table based on the contents of the input record (i.e. *"The matching function uses transaction reference parameter to retrieve the transaction record. Using a reference from the linked list of legacy records, the matching function then retrieves a legacy record...The matching function then performs a comparison of the first field specified in the matching table...When a field matches, the*

positive accumulator is incremented by the score value specified in the matching table for that field... Once all of the fields between two records have been compared, the matching function compares the value of the positive accumulator to a first positive threshold. If the positive accumulator exceeds the positive threshold a match is declared." The preceding text excerpt clearly indicates that data from the matching/index table and the matching function/contents of the reference table are used to compare datum in a field/tokens from the transaction/input record to those of a legacy/reference record in order to select a number of candidate records.) (Column 8, Lines 9-36), and assigning a similarity score to said evaluation data record in relation to a reference record within the reference table (i.e. *"The present invention is a merge or purge system that uses score-based matching condition between records."* The preceding text excerpt clearly indicates that comparisons (e.g. an evaluation data record in relation to a reference record) are assigned scores/similarity scores.) (Column 7, Lines 23-24) based on a combination of: the number of common tokens of an evaluation field of the evaluation data record and a corresponding field within a reference record from the reference table (i.e. *"The equal comparison mechanism compares the fields between two data records for an exact match...The AL comparison mechanism compares two fields for a close alpha match...The letters E and O are treated as identical. The AL compare allows for one transposition of characters."* The preceding text excerpt clearly indicates that each datum contained in a field/token in the legacy/reference records is compared with each datum contained in a field/token in the transaction/evaluation record to find the number of common tokens (e.g. a common token indicates equality). The excerpt also demonstrates that the number of common tokens is determined and utilized in a number of different ways.) (Column 9, Lines 3-4; Column 13, Lines 11-17); and the similarity of the tokens that are not the same in the evaluation field of the evaluation data record and the corresponding field of the reference record from the reference table (i.e. *"The AL comparison mechanism compares two fields for a close alpha match...The letters E and O are treated as identical. The AL compare allows for one transposition of characters."* The preceding text excerpt clearly indicates that non-

identical datum contained in fields/tokens is considered for their similarity (e.g. E and O are considered equivalent).) (Column 13, Lines 11-17).

Lepien fails to disclose evaluating each token with a function to build a vector of token substrings that serve as a signature of the token, each entry of the index table contains a position within the vector, a token substring and a list of records contained within the reference table, and a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table.

Califano discloses evaluating each token with a function to build a vector of token substrings that serve as a signature of the token (i.e. *"The method starts by selecting an original string from a database. The string is then partitioned into substrings of contiguous tokens...a number of original substrings of contiguous tokens are selected from an original token sequence in the database...The set members can all be a fixed number of tokens in length...Using this set of original substring tokens, a set of tuples is formed..."* The preceding text excerpt clearly indicates that the original strings/reference tokens are broken up into substrings which is then represented by a tuple/vector of token substrings that serve as a signature of the token.) (Column 5, Lines 32-35; Column 3, Lines 31-46); and each entry of the index table contains a position within the vector (i.e. *"...More preferably, the information record will also contain information about the location on the original string of the original tuple. Even more preferably, additional information about the tuple and location of the substrings appended to generate the tuple will be included in the cell."* The preceding text excerpt clearly indicates that location information about the tuples/vectors and location information/the position within the vector for the substrings used to create those tuples/vectors is included in the index.) (Column 9, Lines 40-47).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Califano to include evaluating each token with a function to build a vector of token substrings that serve as a signature of the token and each entry of the index table contains a position within the vector with the motivation to provide an improved method for finding sequences of tokens identical or similar to a reference sequence of tokens in one or more original strings of tokens within a database having one or more original strings (Califano, Column 2, Lines 47-49).

Ananthakrishna discloses each entry in the index table including a token substring and a list of records contained within the reference table (i.e. *"We build a token table of G containing the following information...the set of tokens whose frequency...is greater than one...the list of (pointers to) tuples in which such a token occurs..."* The preceding text excerpt clearly indicates that the token/lookup table includes a list of tuples/vectors in which the token/substring occurs (e.g. which maps to the token/substring).) (Page 6, Column 2, Paragraph 6).

It would have been obvious of one skilled in that art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Ananthakrishna to include each entry in the index table including a token substring and a list of records contained within the reference table with the motivation to detect and eliminate duplicated data to improve the area of data cleaning (Abstract; Page 1, Paragraph 2).

Kephart discloses a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table. (i.e. *"As discussed by Salton et al., direct comparison of the document's token frequencies with the token frequencies of each category can lead to highly inaccurate categorization*

because it tends to over-emphasize frequently occurring words such as "the" and "about." This problem is typically avoided by first converting the category token frequencies into category token weights that de-emphasize common words using the Term Frequency-Inverse Document Frequency (TF-IDF) principle. The TF-IDF weight for a token in a specific category increases with the frequency of that token among documents known to belong to the category and decreases with the frequency of that token within the entire collection of documents. There are many different TF-IDF weighting schemes. Salton et al. describe several weighting schemes and their implementations." The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field, which is based on the number of times the token appears in the category/reference data.) (Figure 10; Column 3, Lines 44-58).

It would have been obvious to one skilled in the art at the time of Applicants invention to modify the teachings of Lepien with the teachings of Kephart to include a weight of the tokens of the evaluation data record that is based on a count of the tokens from a corresponding field contained within the reference table with the motivation of assisting a user with the task of categorizing a received electronic document into a collection (Kephart, Abstract).

As per Claim 43 Lepien discloses reference records having a similarity score greater than a threshold are identified as candidate records (i.e. *"Once all of the fields between two records have been compared, the matching function compares the value of the positive accumulator to a first positive threshold. If the positive accumulator exceeds the positive threshold a match is declared. If a definitive match is not found; then the matching function compares the value stored in the negative accumulator to a second negative threshold. If the negative accumulator exceeds the negative threshold a mismatch is declared."* The preceding text excerpt clearly indicates that legacy/reference records, which have a match/similarity score which exceeds/is greater than a threshold, are identified as matches/candidates.) (Column 8, Lines 32-42).

As per Claim 45, Lepien discloses the tokens in different attribute fields are assigned different weights in determining said score (i.e. *"Because the matching system relies on scores, an implicit weighting of certain fields in the records can be helpful in confirming otherwise less than certain matches."* The preceding text excerpt clearly indicates that a weight is assigned to the comparison data for a particular field (e.g. the tokens in the attribute field), and that the weights of different fields may vary.) (Column 7, Lines 29-31).

Allowable Subject Matter

8. Claims 8, 12, 27, 30, 41, and 44 allowed. The following is a statement of reasons for the indication of allowable subject matter:

As per Claims 8, 27, and 41, the prior art neither teaches nor suggests the element of stopping the search for candidate records once a candidate record of a certain degree of similarity has been found.

As per Claims 12, 30, and 44, the prior art of record neither teaches nor suggest the use of a token frequency cache for the purposes of assigning weights to tokens.

9. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Points of Contact

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Hicks whose telephone number is (571) 272-2670. The examiner can normally be reached on Monday - Friday 8:30a - 5:00p.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey Gaffin can be reached on (571) 272-4146. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Michael J Hicks

Art Unit 2165
(571) 272-2670

JEFFREY GAFFIN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100